REMARKS

I. Introduction

With the addition of new claims 15 to 18 and the cancellation herein without prejudice of claim 14, claims 7 to 13, 15 to 18 are currently pending in the present application, since claims 1 to 6 were previously canceled. In view of the foregoing amendments and following remarks, it is respectfully submitted that all of the presently pending claims are allowable, and reconsideration is respectfully requested.

II. Objections to Claim 14

Claim 14 has been canceled herein without prejudice, thereby rendering moot the present objections. Withdrawal of these objections to claim 14 is therefore respectfully requested.

III. Rejection of Claims 7 to 14 Under 35 U.S.C. § 112, Second Paragraph

Claims 7 to 14 were rejected under 35 U.S.C. § 112, second paragraph, as allegedly indefinite. It is respectfully submitted that the pending claims are definite for at least the following reasons.

As an initial matter, claim 14 has been canceled herein without prejudice, thereby rendering moot the present rejection with respect to claim 14.

As for claims 7 to 13, while the rejection may not be agreed with, to facilitate matters, claim 7 has been rewritten to state "a reconfigurable field of data processing cells," as suggested by the Office Action at page 3.

As for claim 11, the Office Action asserts that the limitations of "the program includes a multitask application" and "one of two different tasks of the multitask application" are indefinite. It is respectfully submitted that one of ordinary skill in the art would recognize the feature of "the program includes a multitask application" as referring to a computer program and the like associated with multiple tasks that are performed via multitasking, and the feature of "one of two different tasks of the multitask application" as referring to tasks of the computer program which includes the multiple tasks. In addition, "multitasking" is further described in the Substitute Specification at page 15, line 22 to page 17, line 12. Therefore, it is respectfully submitted that claim 11 is definite.

It is therefore respectfully submitted that the claims are clear, give rise to no ambiguity, and, therefore, definite. In view of all of the above, withdrawal of this rejection is respectfully requested.

IV. Rejection of Claims 7 to 10 Under 35 U.S.C. § 103(a)

Claims 7 to 10 were rejected under 35 U.S.C. § 103(a) as unpatentable over the combination of International Patent Application Publication No. WO 99/40522 ("Gonion et al."), U.S. Patent No. 5,941,977 ("Panwar et al."), and U.S. Patent No. 6,374,286 ("Gee et al."). It is respectfully submitted that the combination of Gonion et al., Panwar et al., and Gee et al. does not render unpatentable any of claims 7 to 10 for at least the following reasons.

In order for a claim to be rejected for obviousness under 35 U.S.C. § 103(a), the prior art must teach or suggest each element of the claim. *See Northern Telecom, Inc. v. Datapoint Corp.*, 908 F.2d 931, 934 (Fed. Cir. 1990), *cert. denied*, 111 S. Ct. 296 (1990); *In re Bond*, 910 F.2d 831, 834 (Fed. Cir. 1990). To establish a *prima facie* case of obviousness, the Examiner must show, *inter alia*, that there is some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify or combine the references, and that, when so modified or combined, the prior art teaches or suggests all of the claim limitations. M.P.E.P. §2143. In addition, as clearly indicated by the Supreme Court, it is "important to identify a reason that would have prompted a person of ordinary skill in the relevant field to combine the [prior art] elements" in the manner claimed. *See KSR Int'l Co. v. Teleflex, Inc.*, 127 S. Ct. 1727 (2007).

Claim 7, as presented, relates to a method of data processing using a processor including a reconfigurable field of data processing cells and a register, and, as herein amended without prejudice, recites the following:

> *. . . providing a program corresponding to a sequence of instructions to the processor for execution of the program;*
> *determining, for the reconfigurable field of data processing cells, a configuration set corresponding to the program and by running of which the program is executed, the configuration set including a sequence of configurations;*
> *determining, for each configuration, a respective maximum allowed execution runtime prior to lapse of which the respective configuration is uninterruptible;*
> *executing the sequence of configurations; and*
> *during the executing:*
> *storing, in the data stream memory, at least one of the data stream and parts of the data stream; and*
> *for each configuration, monitoring the respective maximum allowed execution runtime in order to interrupt the configuration if the respective maximum allowed execution runtime is exceeded.*

Support for the amendments to the claims may be found in the Substitute Specification, e.g., at page 7, lines 5 to 7; page 8, lines 10 to 23; and page 20, lines 8 to 10.

The Office Action admits, at page 5, that "both Gonion and Panwar do not disclose determining, for each configuration, a respective maximum allowed execution runtime prior to lapse of which the respective configuration is uninterruptible."

Further, Gee et al. do not correct this deficiency of the combination of Gonion et al. and Panwar et al., since Gee et al. do not disclose or suggest the features of *determining a respective maximum allowed execution runtime for each of a sequence of configurations of a configuration set that corresponds to a program*, as provided for in the context of claim 7.

These features of claim 7 are not rendered obvious by prior art referring to the use of a reconfigurable device, by prior art referring to sequential processors, or by their combination.

In this regard, it is noted that a program for a conventional processor is defined by instructions. An instruction for a conventional processor can never exceed a given time. Even if a user experiences the problem of a program that is not responding, the processor itself will nevertheless continue to execute instructions, although this might not lead to a useful or noticeable result of, e.g., a PC being used. Accordingly, there has never been a need to provide a time period for an individual instruction during which it cannot be interrupted but after which it can be interrupted. Indeed, this has never been done.

Instead, for a conventional processor, the time needed for the execution of a number of applications as a whole has been monitored.

For example, Gee et al. refer to operating multiple Java Virtual Machines (JVMs) in separate time slices (partitions) on a single processor, where one master JVM controls the transfers between different JVMs. One JVM of Gee et al. executes a number of applications and threads, and each application or thread will consist of a very large number of instructions. Gee et al. merely indicate that when one JVM (executing a number of applications or threads) exceeds its allotted time, it is terminated. Thus, Gee et al. merely allot fixed periods of time to each JVM that runs plural applications (each consisting of a number of instructions) and threads, but do not determine fixed periods of time for each instruction of the applications. Indeed, as explained above, since a given instruction cannot exceed a certain amount of time, there would be no use in doing so.

This is important under certain conditions, e.g., when debugging a program. Termination of an application might indicate a programming error, e.g., due to an infinite loop. The programmer that encounters such a situation has the possibility to step through such an application such that a single instruction is executed, the state of the processor is

saved and then the next single instruction is executed. Accordingly, the programmer looks at a series instructions and their corresponding states. Here, each instruction will be executed in a given time and thereafter, the state can be examined. This can be done as a reaction to an application consisting of plural instructions stalling.

Having noted this, it is clear that providing a maximum allowed execution runtime prior to lapse of which a respective configuration (of a set that corresponds to a program, *i.e.*, where the configurations correspond to instructions of the program) is uninterruptible, as provided for in the context of claim 7, cannot be compared to the termination in Gee et al. of a JVM (executing a number of applications or threads) when it exceeds its allotted time.

As stated above, when an application is stalling, the programmer might look at the single instructions of the application to identify the fault. If a configuration would be analogous to an application, what instructions or finer detail could the programmer look at to debug it? There are no parts of the configuration that can be looked at in the same manner as the instruction for a conventional processor. After the execution of an application for a given time there will usually not be a defined state. So, if a configuration would be analogous to an application, what state would the programmer have to look at to debug it?

Accordingly, treating the configurations in an application-like manner will prevent debugging. Only when it is understood that -- contrary to the assumption of the examiner -- a configuration does not correspond to the entire series of instructions forming an application but rather to a single instruction, will debugging become possible.

Yet, even if it is understood by the average skilled person that configurations should correspond to instructions rather than applications, it must still be doubted that the average skilled person would see the need to define a maximum run time for such a configuration. First, this is not necessary for an instruction in a conventional processor. Further, even if the average skilled person would realize that it would be advantageous for a configuration considered in a manner analogous to an instruction to have a restricted time, the average skilled person would still not end up in a more favorable situation with regard to, e.g., debugging, as there would still not be access to something corresponding to state information in a conventional processor. So, there seems to be no use in restricting the run time because there still is no way of debugging the program. Debugging the program becomes possible only if, in addition to the time restriction to the configuration, some state-like information is made available as well. This is done by storing, in the data stream

memory, at least one of the data stream and parts of the data stream; thus treating the memory like a register.

Thus, Gee et al. do not disclose, or suggest, the features of *determining a respective maximum allowed execution runtime for each of a sequence of configurations that corresponds to a program*, as provided for in the context of claim 7, as presented. Further, one skilled in the art would not have been motivated to modify Gee et al. to include these features since one skilled in the art would not have determined any advantage in restricting, e.g., the single instruction.

Thus, the combination of Gonion et al., Panwar et al., and Gee et al. does not disclose or suggest all of the features included in claim 7, as presented, so that the combination of Gonion et al., Panwar et al., and Gee et al. does not render unpatentable claim 7.

Claims 8 to 10 ultimately depend from claim 7 and are therefore allowable for at least the same reasons set forth above in support of the patentability of claim 7. *In re Fine*, 837 F.2d 1071, 5 U.S.P.Q.2d 1596 (Fed. Cir. 1988) (any dependent claim that depends from a non-obvious independent claim is non-obvious).

Withdrawal of this obviousness rejection is therefore respectfully requested.

V. Rejection of Claims 11 and 14 Under 35 U.S.C. § 103(a)

Claims 11 and 14 were rejected under 35 U.S.C. § 103(a) as unpatentable over the combination of Gonion et al., Panwar et al., Gee et al., and U.S. Patent No. 5,860,119 ("Dockser"). It is respectfully submitted that the combination of Gonion et al., Panwar et al., Gee et al., and Dockser does not render unpatentable the present claims for at least the following reasons.

As an initial matter, claim 14 has been canceled herein without prejudice, thereby rendering moot the present rejection with respect to claim 14.

Claim 11 depends from claim 7 and is therefore allowable for at least the same reasons set forth above in support of the patentability of claim 7 since Dockser does not cure the deficiencies noted above with respect to the combination of Gonion et al., Panwar et al., and Gee et al.

Withdrawal of this obviousness rejection is therefore respectfully requested.

VI. Rejection of Claims 12 and 13 Under 35 U.S.C. § 103(a)

Claims 12 and 13 were rejected under 35 U.S.C. § 103(a) as unpatentable over the combination of Gonion et al., Panwar et al., Gee et al., and U.S. Patent No. 4,041,462 ("Davis et al."). It is respectfully submitted that the combination of Gonion et al., Panwar et al., Gee et al., and Davis et al. does not render unpatentable either of claims 12 and 13 for at least the following reasons.

Claims 12 and 13 ultimately depend from claim 7 and are therefore allowable for at least the same reasons set forth above in support of the patentability of claim 7 since Davis et al. do not cure the deficiencies noted above with respect to the combination of Gonion et al., Panwar et al., and Gee et al.

Withdrawal of this obviousness rejection is therefore respectfully requested.

VII. New Claims 15 to 18

New claims 15 to 18 have been added herein. It is respectfully submitted that new claims 15 to 18 do not add any new matter and are fully supported by the present application, including the Substitute Specification, e.g., at page 8, lines 10 to 23. Claims 15 to 18 ultimately depend from claim 7 and are therefore allowable for at least the same reasons set forth above in support of the patentability of claim 7.

VIII. Conclusion

In light of the foregoing, it is respectfully submitted that all of the presently pending claims are in condition for allowance. Prompt reconsideration and allowance of the present application are therefore earnestly solicited.

Respectfully submitted,

Dated: January 24, 2008

By: /_____/

Aaron Grunberger
Reg. No. 59,210

KENYON & KENYON LLP
One Broadway
New York, New York 10004
(212) 425-7200

CUSTOMER NO 26646